4.2. Science Operations Activities

4.2.1 Science Data Ingest Activities

Science data ingest activities at ECS DAACs include data transfer, basic metadata checking, conversion, and storage of data in the appropriate data server. The different modes of ingest supported by the Ingest Client software are further defined in the following scenarios. The objective of the following paragraphs is to demonstrate that the entire data ingest process is largely automated. DAAC staff will, however, be required to support hard media operations, resolve problems, periodically monitor ingest operations, and coordinate with the appropriate internal and external entities to resolve resource conflicts.

4.2.1.1 Automated Network Ingest Scenario

This scenario describes the automated network ingest of data to ECS from data providers which will be accomplished without direct operator action. Figure 4.2.1.1-1 and Table 4.2.1.1-1 depict the steps involved in this scenario. Examples of data providers that will use this interface for transfer of data to ECS include the Sensor Data Processing Facility (SDPF), and TRMM Science Data and Information System (TSDIS). An authentication request is sent from the originating system to the destination system to solicit verification of the originating system as a valid user. At various stages during the asynchronous handshaking procedure, authentication of either ECS or the data provider is required, depending on the particular data transfer message to be sent. The authentication requests and responses are omitted from the description of each data transfer message for brevity. The data provider will send a Data Availability Notice (DAN) to the Ingest Subsystem indicating that data is ready for transfer. The DAN identifies parameters such as data source, number of files, and location of data, and a summary of the DAN contents is placed in the event log. The Ingest Subsystem generates a corresponding ingest request and stores the request on a prioritized list. A Data Availability Acknowledgment (DAA) is sent from Ingest to the data provider indicating readiness to ingest the data identified in the DAN. The ingest function ensures that all required devices are allocated and schedules and performs the data transfer. Data transfer status, including all recoverable errors, is indicated in the event log. Once the data transfer is complete, the ingest function extracts the metadata and checks selected metadata parameters (e.g., header information). The status of the metadata parameter check is written to the event log. The ingest function then generates a data server insert request to store the data and metadata in the appropriate data repository. Subscriptions (if any) are triggered to indicate the availability of data once the archive process is completed. A Data Delivery Notice (DDN) is sent to the data provider indicating that the archiving of the data identified in the DAN has been completed. The data provider returns a Data Delivery Acknowledgment (DDA) in response to the DDN and terminates the session. Ingest provides a status message to the Ingest History Log when the transaction is complete. All messages transferred between subsystems and between processes within the Ingest Subsystem are sent to log files which may be monitored or viewed by operations personnel.

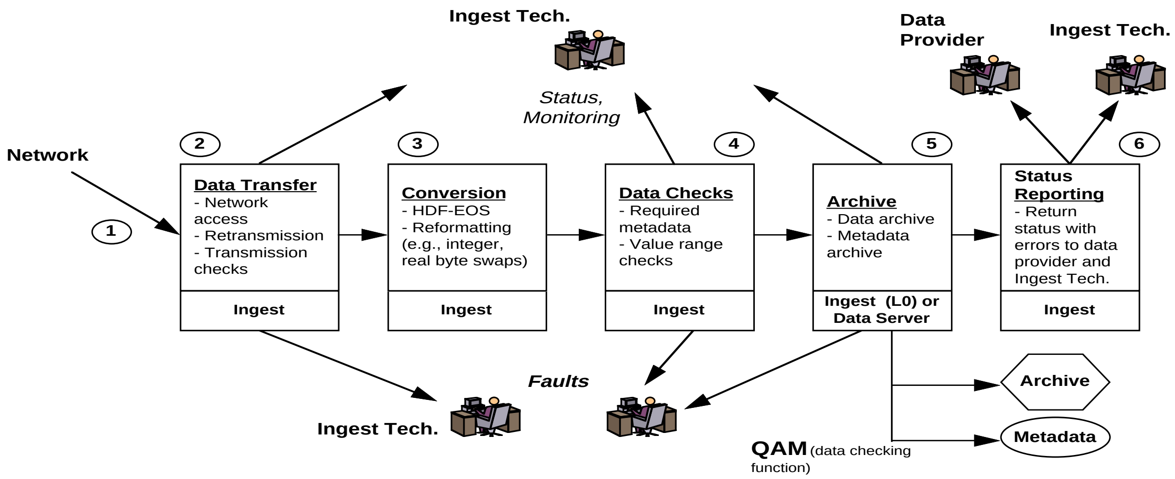


Figure 4.2.1.1-1. Automated Network Ingest Scenario

Table 4.2.1.1-1. Automated Network Ingest Scenario (1 of 2)

Purpose and Precondition:

Automated Network Ingest requires that the ingest function always be ready to receive and process an incoming DAN. The two polling mechanisms require that the ingest client software for those interfaces always be active to some degree and periodically and automatically check a specified location for data.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Ingest Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files.	Data provider sends DAN to Ingest. Receipt of the DAN is logged. This process is assigned a request ID, and from this point forward the event log and status display will contain information related to this transaction. Ingest generates a corresponding ingest request and stores the request on a prioritized list. A summary of the DAN contents indicating data source, number of files, and location of data is placed in the event log. The system acknowledges the request with a DAA, and a copy of the DAA indicating readiness to ingest data is logged	Initiate session between data provider and ECS Ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests.	The ingest function schedules and performs data transfer. Devices allocated to the data transfer are identified. Data transfer status (including recoverable errors) are indicated in the event log.	Transfer data from data provider to ECS Ingest.

Table 4.2.1.1-1. Automated Network Ingest Scenario (2 of 2)

Step	Operator/User	System	Purpose
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.
4		Ingest function extracts metadata parameters. Status of metadata parameter check is written to event log.	Validation of select metadata parameters.
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	A DDN is sent to the data provider and the DDA is logged when received. Status messages are provided to the data provider and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.2 Polling Ingest with Delivery Record Scenario

The following scenario describes the manner in which data is ingested into ECS using the polling ingest with delivery record mechanism. Figure 4.2.1.2-1 and Table 4.2.1.2-1 depict the steps involved in this scenario. This mechanism is planned to be used for the transfer of data from the EDOS Data Processing Facility (DPF) to ECS. An authentication request is sent from the originating system to the destination system to solicit verification of the originating system as a valid user. At various stages during the asynchronous handshaking procedure, authentication of either ECS or the data provider is required, depending on the particular data transfer message to be sent. The authentication requests and responses are omitted from the description of each data transfer message for brevity. The ingest polling client periodically checks an agreed-upon network location for a Delivery Record file. The Delivery Record file contains information similar to that in a DAN, and describes the location of the available data. All data at the specified location are assumed to make up a collection of ingest data with one file per data granule. If a Delivery Record is located, the ingest function generates a corresponding ingest request and stores the request on a prioritized list. If data is located at the specified location, the ingest function automatically performs an ftp get from the source within a system-tunable time window. Format conversion, as required, is performed on the ingested data, as well as metadata extraction and validation. The ingest function then generates a data server insert request to store the data and metadata in the appropriate data repository. Subscriptions (if any) are triggered to indicate the availability of data once the archive process is completed. The polling ingest client resets the polling interval and enters a wait state. Ingest provides a status message to the Ingest History Log when the transaction is complete.

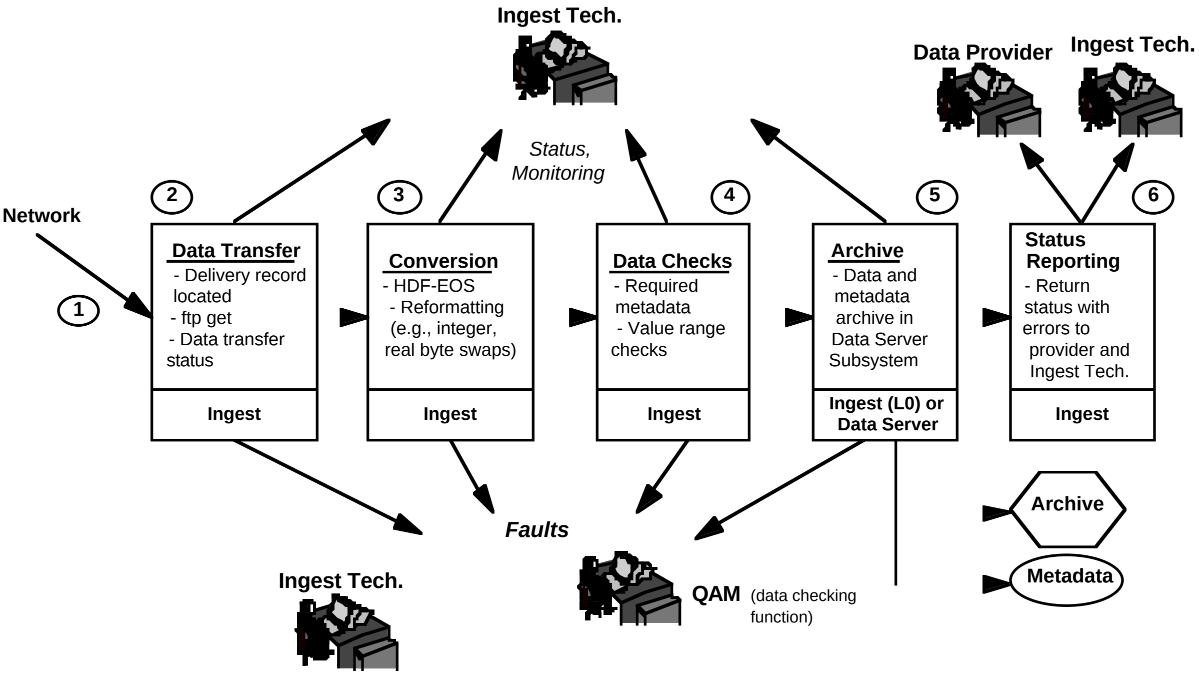


Figure 4.2.1.2-1. Polling Ingest With Delivery Record Scenario

Table 4.2.1.2-1. Polling Ingest With Delivery Record Scenario 1 of 2)

Purpose and Precondition:

Automated Network Ingest requires that the ingest function always be ready to receive and process an incoming DAN. The two polling mechanisms require that the ingest client software for those interfaces always be active to some degree and periodically and automatically check a specified location for data.

Step	Operator/User	System	Purpose
1		The polling ingest client periodically checks an agreed-upon network location for a delivery record. Network address, status (faults), request ID, and start of the ingest process will be indicated in the event log when data is located. Ingest generates a corresponding ingest request and stores the request on a prioritized list.	Initiate session between data provider and ECS Ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests	If a delivery record is located, Ingest automatically performs an ftp get from the source within a system-tunable time window. Data transfer status is indicated in the event log.	Transfer data from data provider to ECS Ingest.
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.

Table 4.2.1.2-1. Polling Ingest With Delivery Record Scenario (2 of 2)

Step	Operator/User	System	Purpose
4		Ingest function extracts metadata parameters. Status of metadata parameter check is written to event log.	Validation of select metadata parameters.
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	The polling ingest client resets the polling interval and enters a wait state. Status messages are provided to the data provider and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.3 Polling Ingest Without Delivery Record Scenario

The following scenario describes the manner in which data is ingested into ECS using the polling ingest without delivery record mechanism. Figure 4.2.1.3-1 and Table 4.2.1.3-1 depict the steps involved in this scenario. This mechanism is planned to be used for the transfer of certain ancillary products required for TRMM data processing. An authentication request is sent from the originating system to the destination system to solicit verification of the originating system as a valid user. At various stages during the asynchronous handshaking procedure, authentication of either ECS or the data provider is required, depending on the particular data transfer message to be sent. The authentication requests and responses are omitted from the description of each data transfer message for brevity. The ingest polling client periodically checks an agreed-upon network location for the presence of data. All data at the specified location are assumed to make up a collection of ingest data with one file per data granule. If data is located at the specified location, the ingest function automatically performs an ftp get from the source within a system-tunable time window. Format conversion, as required, is performed on the ingested data, as well as metadata extraction and validation. The ingest function then generates a data server insert request to store the data and metadata in the appropriate repository. Subscriptions (if any) are triggered to indicate the availability of data once the archive process is completed. The polling ingest client resets the polling interval and enters a wait state. Ingest provides a status message to the Ingest History Log when the transaction is complete.

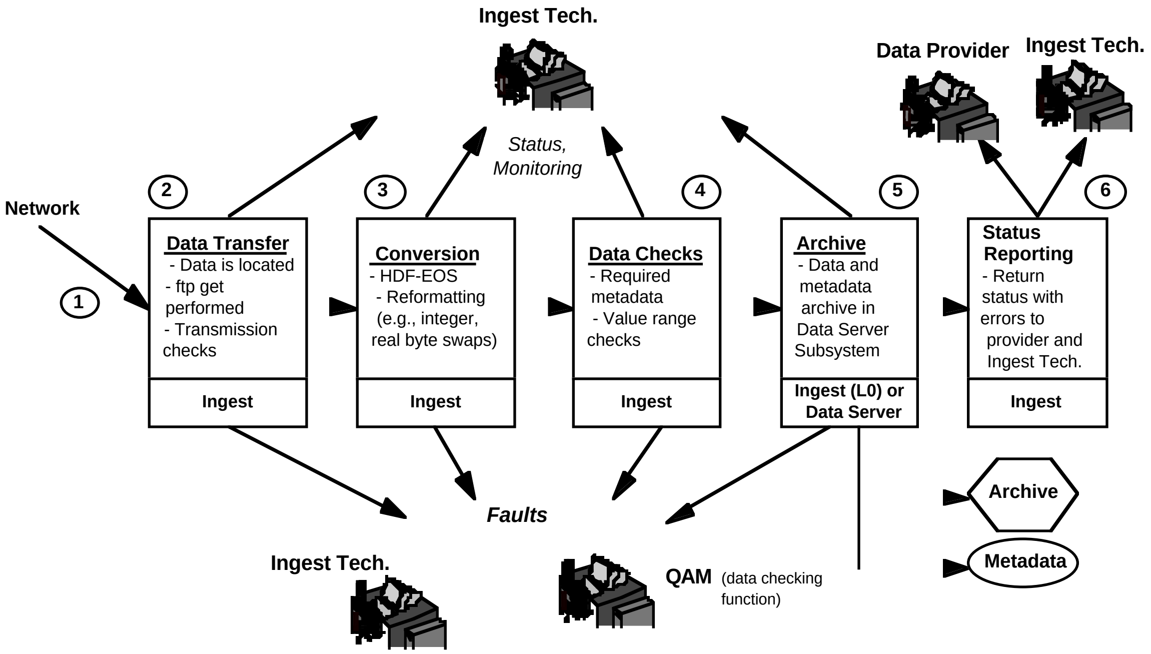


Figure 4.2.1.3-1. Polling Ingest Without Delivery Record Scenario

Table 4.2.1.3-1. Polling Ingest Without Delivery Record Scenario 1 of 2)

Purpose and Precondition:

Automated Network Ingest requires that the ingest function always be ready to receive and process an incoming DAN. The two polling mechanisms require that the ingest client software for those interfaces always be active to some degree and periodically and automatically check a specified location for data.

Step	Operator/User	System	Purpose
1		The polling ingest client periodically checks an agreed-upon network location for available data. Network address, status (faults), request ID, and start of the ingest process will be indicated in the event log when data is located. Ingest generates a corresponding ingest request and stores the request on a prioritized list.	Initiate session between data provider and ECS Ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests	If data is located, Ingest automatically performs an ftp get from the source within a system-tunable time window. Data transfer status is indicated in the event log.	Transfer data from data provider to ECS Ingest.

Table 4.2.1.3-1. Polling Ingest Without Delivery Record Scenario (2 of 2)

Step	Operator/User	System	Purpose
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.
4		Ingest function extracts metadata parameters. Status of metadata parameter check is written to event log.	Validation of select metadata parameters.
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	The polling ingest client resets the polling interval and enters a wait state. Status messages are provided to the data provider and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.4 Hard Media Ingest Scenario

The following scenario describes the manner in which data received on hard media is ingested into ECS. Figure 4.2.1.4-1 and Table 4.2.1.4-1 depict the steps involved in this scenario. Media that is received at the DAAC is checked for readiness to ingest. The ingest technician compares the received media to a media ingest readiness checklist and invokes the ingest client s/w via the GUI once the media has been readied. A device allocation is requested from the Data Server peripheral pool. The ingest technician receives the device id and is prompted to mount the media. The existence of a Delivery Record file describing the media contents is checked and a summary of the Delivery Record contents is logged. The data is transferred from media to working storage, and basic metadata extraction and validation is performed. The ingest function then generates a data server insert request to store the data and metadata in the appropriate data repository. Subscriptions (if any) are triggered to indicate the availability of data once the archive process is completed. Email notification of successful ingest is sent to the data provider if a network address is available.

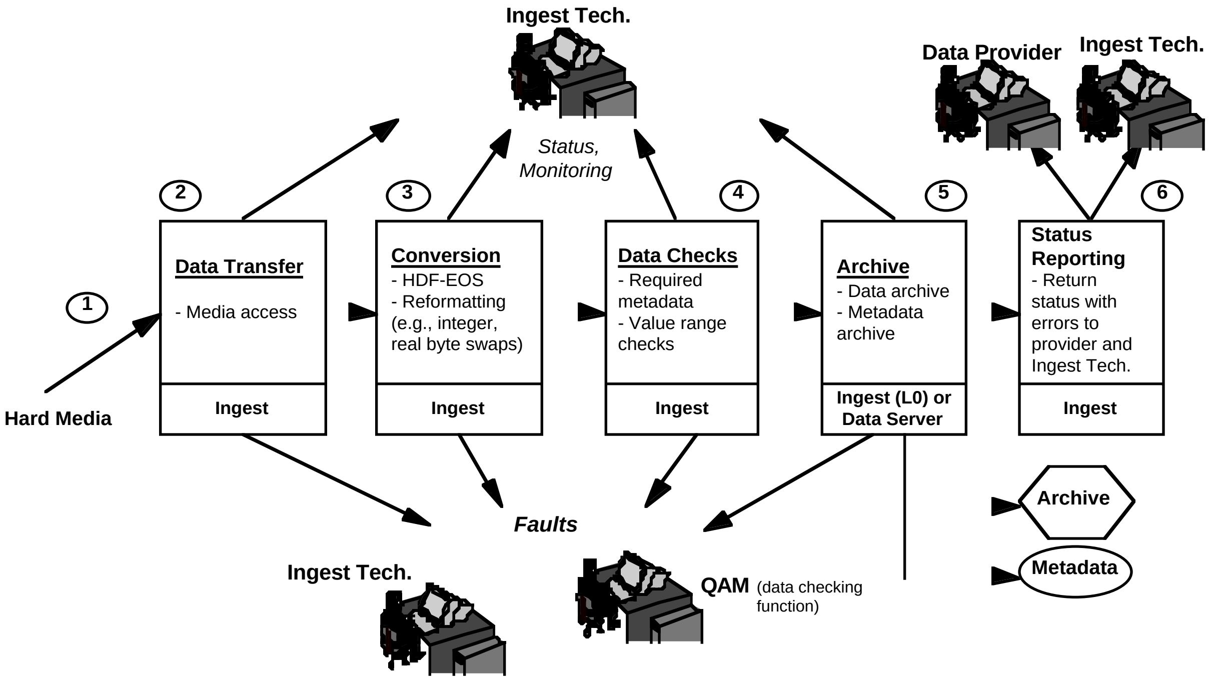


Figure 4.2.1.4-1. Hard Media Ingest Scenario

Table 4.2.1.4-1. Hard Media Ingest Scenario (1 of 2)

Purpose and Precondition:

The only operator-related precondition during normal operations deals with the physical handling and checking of media that must be performed prior to the start of hard media ingest.

Step	Operator/User	System	Purpose
1	The Ingest Technician compares the received media against the media ingest readiness checklist. The Ingest Technician then invokes the ingest client s/w via the GUI.		Hard media is received at the DAAC and is checked for readiness to ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests. The Ingest Technician receives the device ID and mounts the media when prompted to do so.	A device allocation is requested from the Data Server peripheral pool. The existence of a Delivery Record file is checked and a summary of its contents is logged. The Ingest Technician is prompted to mount the media. Data is transferred from media to working storage. Status of all involved devices is written to the event log. Peripherals are deallocated when data transfer is complete.	Transfer data from media to ECS Ingest.
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.

Table 4.2.1.4-1. Hard Media Ingest Scenario (2 of 2)

Step	Operator/User	System	Purpose
4		Ingest function extracts metadata parameters. Status of metadata parameter check is written to event log.	Validation of select metadata parameters.
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	Email notification is sent to the data provider if a network address is available. Status messages are provided to the Ingest Technician and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.5 Interactive Network Ingest Scenario

The User Network Ingest Interface CSC provides authorized ECS science users with the capability to interactively request network ingest data into the ECS system. Figure 4.2.1.5-1 and Table 4.2.1.5-1 depict the steps involved in this scenario. This process is initiated when a user selects the User Network Ingest option from the ECS GUI screens. The user retrieves the Ingest Request Form via HyperText Markup Language (HTML) and fills in the fields appropriate to the type of data being ingested. The user submits the completed form via HTTP, and the Ingest HTTP daemon invokes the Ingest Form Script. Upon invocation, the Ingest Form Script will package the Request Form fields into a DAN message and sends the message to ECS Ingest for processing. Once Ingest processes the DAN, the data transfer and system and operator actions correspond to those described in the Automated Network Ingest Scenario (section 4.2.1.1).

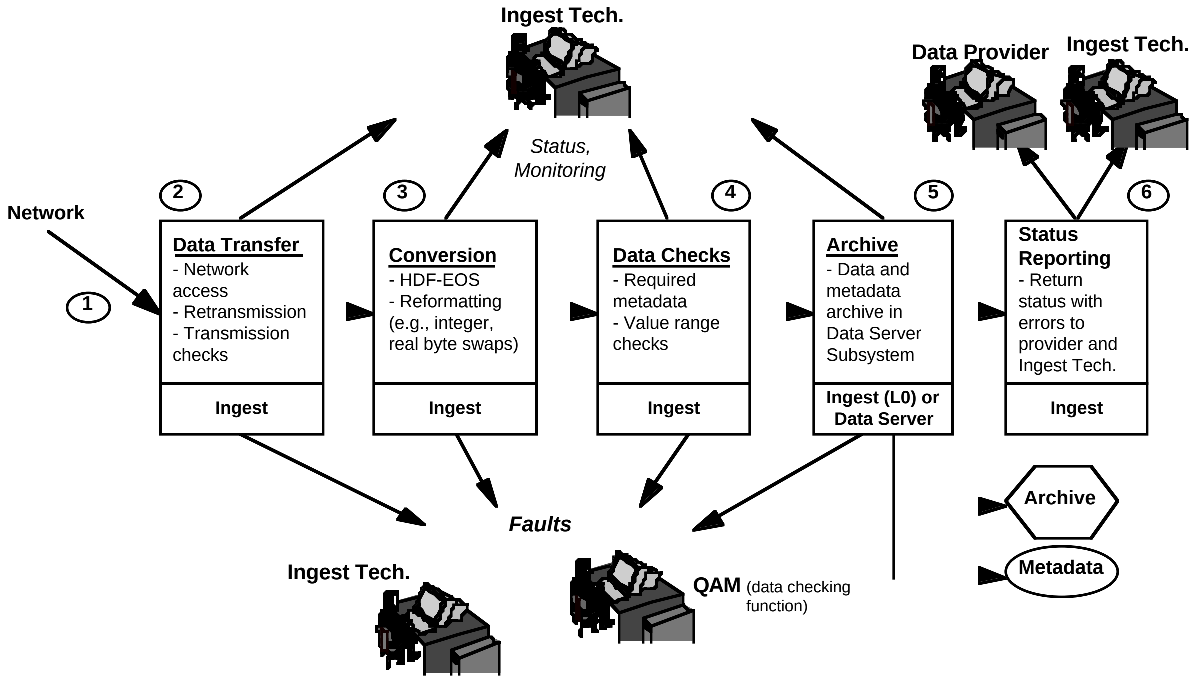


Figure 4.2.1.5-1. Interactive Network Ingest Scenario

Table 4.2.1.5-1. Interactive Network Ingest Scenario (1 of 2)

Step	Operator/User	System	Purpose
1	The user selects the "User Network Ingest" option from the available GUI screens.	The Request Form screen is available for the user to fill in with the appropriate information. The filled-in fields in the Request Form are parsed by the HTTP daemon. Upon invocation, the Ingest Form Script repackages the Request Form fields into a DAN message and sends the message to ECS Ingest for processing.	Identify what data the user wishes to enter into ECS.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests.	The ingest function schedules and performs data transfer. Devices allocated to the data transfer are identified. Data transfer status (including recoverable errors) are indicated in the event log.	Transfer data from data provider to ECS Ingest.
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.
4		Ingest function extracts metadata parameters. Status of metadata parameter check is written to event log.	Validation of select metadata parameters.

Table 4.2.1.5-1. Interactive Network Ingest Scenario (2 of 2)

Step	Operator/User	System	Purpose
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	A DDN is sent to the data provider and the DDA is logged when received. Status messages are provided to the data provider and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.6 Version 0 Data Ingest Scenario

The following scenario describes the mechanism for ingest of Version 0 data. Figure 4.2.1.6-1 and Table 4.2.1.6-1 depict the steps involved in this scenario. Version 0 data preparation, format conversion, etc. will be accomplished by the V0 migration facility. This facility consists of a string of hardware and software at each DAAC that supports the preparation of Version 0 data into a form that may be ingested by the standard ingest client software. The electronic ingest scenario for the ingest of Version 0 data subsequent to preparation by the Version 0 migration facility is similar to that for the automated network ingest scenario. The ingest of Version 0 data via hard media is similar to the hard media ingest scenario.

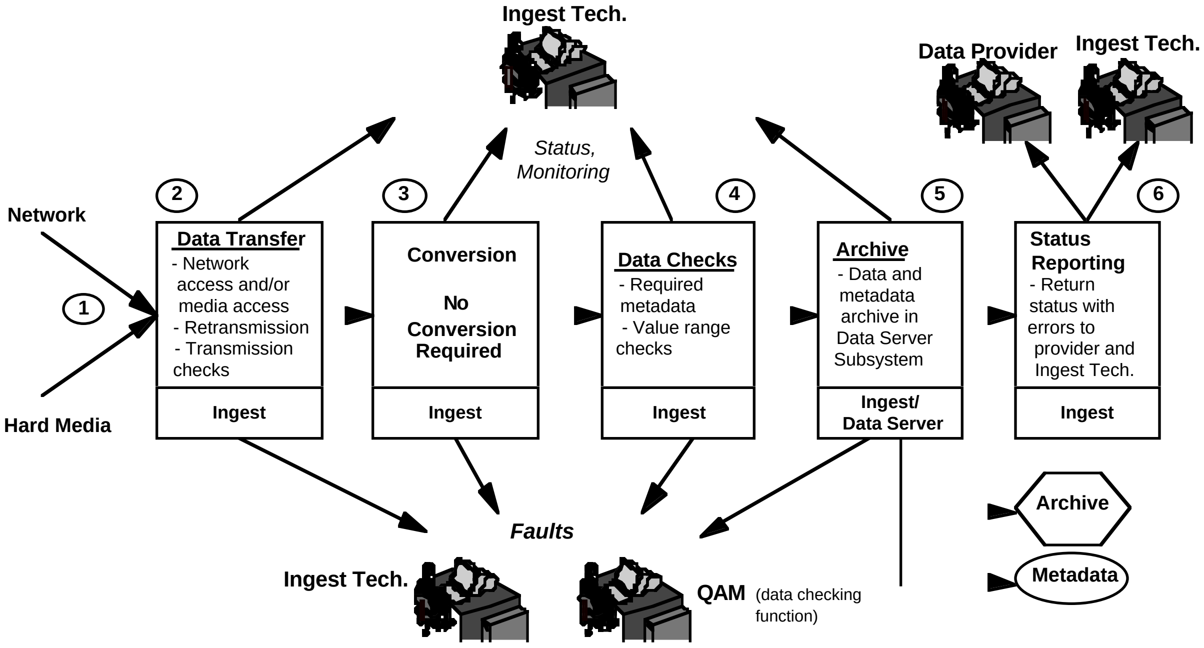


Figure 4.2.1.6-1. Version 0 Data Ingest Scenario

Table 4.2.1.6-1. Version 0 Data Ingest Scenario

Purpose and Precondition:

Version 0 data preparation, format conversion, etc. will be accomplished by the V0 migration facility.

Step	Operator/User	System	Purpose
1		Version 0 data preparation, format conversion, etc. will be accomplished prior to data ingest by the V0 migration facility. Upon data readiness it sends a DAN to Ingest or provides hard media with a Delivery Record file. From this point on, the steps taken are the same as either those listed for Automated Network Ingest (Table 4.2-1) or for Hard Media Ingest (Table 4.2-4). The steps below assume use of the Automated Network Ingest Scenario.	Initiate session between data provider and ECS Ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests.	The ingest function schedules and performs data transfer. Devices allocated to the data transfer are identified. Data transfer status (including recoverable errors) are indicated in the event log.	Transfer data from migration facility to ECS Ingest.
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.
4		Ingest function extracts metadata parameters. Status of metadata parameter check is written to event log.	Validation of select metadata parameters.
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	A DDN is sent to the data provider and the DDA is logged when received. Status messages are provided to the data provider and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.7 Bad Data Scenario

This scenario injects some possible fault conditions into the nominal Automated Network Ingest Scenario (Section 4.2.1.1) to show system and operator actions taken in response to receipt of bad data during ingest. Figure 4.2.1.7-1 provides a pictorial illustration of this scenario, and Table 4.2.1.7-1 depicts the sequence of events involved in this scenario. It should be noted that the Ingest Subsystem provides highly automated ingest of known data products (possibly excluding certain User Network Ingest cases) from known data providers. The instances of Ingest receiving data that is completely unknown or unreadable should be rare. Therefore, the more likely bad data scenarios for ingest include:

- An error is made by the data provider in the generation of the DAN or Delivery Record incorrectly or incompletely specifying the data to be ingested
- A network or device failure causes a glitch in the data as it is being transferred
- Some anomaly is present in the data itself (e.g., parameters out of range)

The first example of a fault condition is the receipt of an incomplete or damaged DAN from the data provider. If the DAN is unreadable, the ingest software automatically returns a DAA indicating that the DAN was invalid and requests a retransmission. If the DAN is readable, but does not match the metadata parameters of the data once the data is ingested, the Data Specialist will be notified of the problem. In most cases, archiving of the data will continue, since the existence of an out of range metadata parameter may not necessarily indicate that the data is faulty. If the data is indeed faulty, the data provider will be notified of the problem and the data provider will be required to request reingestion of the data once the problem has been resolved.

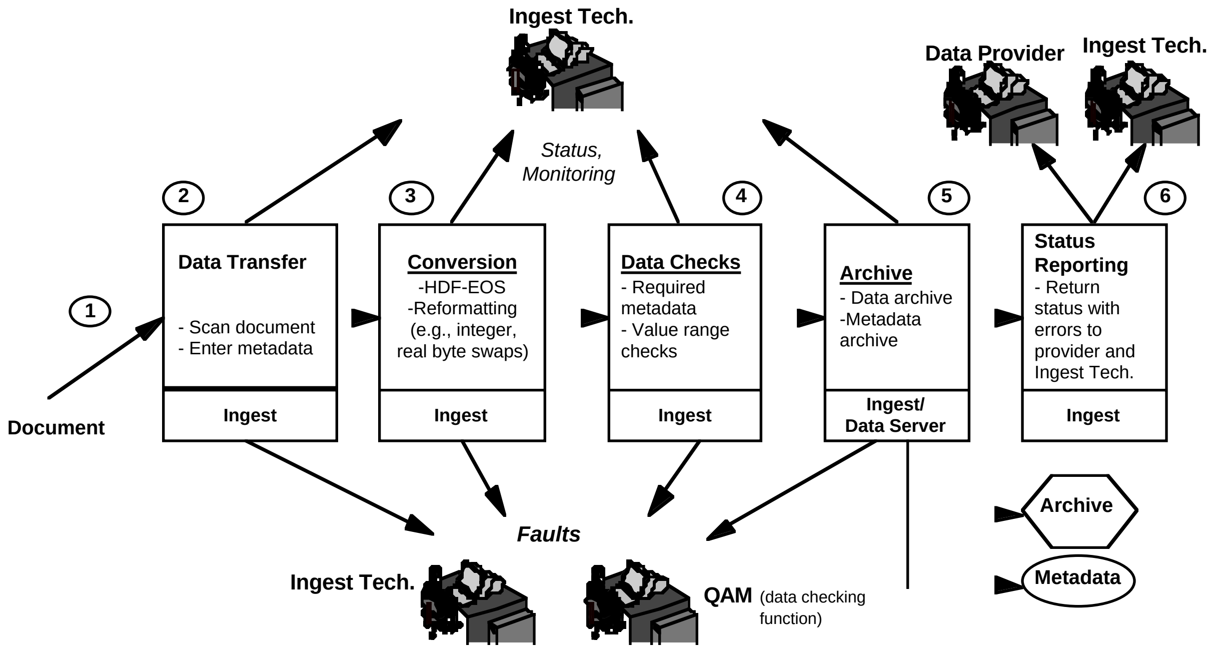


Figure 4.2.1.7-1. Bad Data Scenario

Table 4.2.1.7-1. Bad Data Scenario

Purpose and Precondition:

The ingest function is ready to receive and process an incoming Data Availability Notice (DAN). A similar scenario occurs for the Ingest polling interfaces--polling with and without delivery record. The two polling mechanisms require that the ingest client software for those interfaces be active and periodically check a specified location for data.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Ingest Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the event log files.	Data provider sends a Data Availability Notice (DAN) to Ingest. Receipt of the DAN is logged. This process is assigned a request ID, and from this point forward the event log and status display will contain information related to this transaction. Ingest generates a corresponding ingest request and stores the request on a prioritized list. If an error in the DAN is identified (e.g., invalid data provider, missing file name field), the system returns a Data Availability Acknowledgement (DAA) to the data provider indicating the DAN error, and a status message is logged.	Initiate session between data provider and ECS Ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests.	When a valid DAN is received, the ingest function schedules and performs data transfer. Devices allocated to the data transfer are identified. Data transfer status (including recoverable errors) are sent to the event log. During transfer, a network outage occurs, halting data transfer. The system will automatically try to recover for an operator-tunable number of times. If the connection cannot be established, the session is terminated. A trouble ticket is generated, and a Data Delivery Notice (DDN) indicating transfer failure is sent to the data provider as soon as the network connection is reestablished.	Transfer data from data provider to ECS Ingest.
3		Perform data conversion or reformatting as required for the particular data being ingested. If an error is detected, a DDN is returned to the data provider indicating the error. Status messages are sent to the event log.	Convert data to ECS-supported format.
4		Ingest function extracts metadata parameters. If a metadata parameter is outside of the specified range, a DDN is returned to the data provider indicating the error. Status of the metadata parameter check is sent to the event log.	Validation of select metadata parameters.
5		Ingest function generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display after a time delay.	A DDN is sent to the data provider indicating successful/unsuccessful storage of the data, and the Data Delivery Acknowledgement (DDA) is logged when received from the data provider. Summary statistical information is stored in the Ingest History Log.	Completion of ingest session.

4.2.1.8 Document Ingest Scenario

This scenario describes the ingest of data to ECS from data providers through the scanning and digitizing of hard copy documents. Figure 4.2.1.8-1 provides a pictorial illustration of this scenario, and Table 4.2.1.8-1 depicts the sequence of events involved in this scenario. Documents that are received at the DAAC are checked to ensure that they are from authorized data providers and that the required metadata (e.g., document name, author) has been provided. The ingest technician invokes the ingest client s/w via the GUI once the document is ready for ingest. A device allocation is requested from the Data Server peripheral pool. The ingest technician receives the device id and is prompted to insert the document in the appropriate scanner. The document is scanned and the digitized data is transferred to working storage. Using the Ingest GUI the Ingest Technician enters the basic metadata for the scanned document. The Ingest Technician then generates a data server insert request to store the data and metadata in the appropriate data repository. Subscriptions (if any) are triggered to indicate the availability of data once the archive process is completed. Email notification of successful ingest is sent to the data provider if a network address is available.

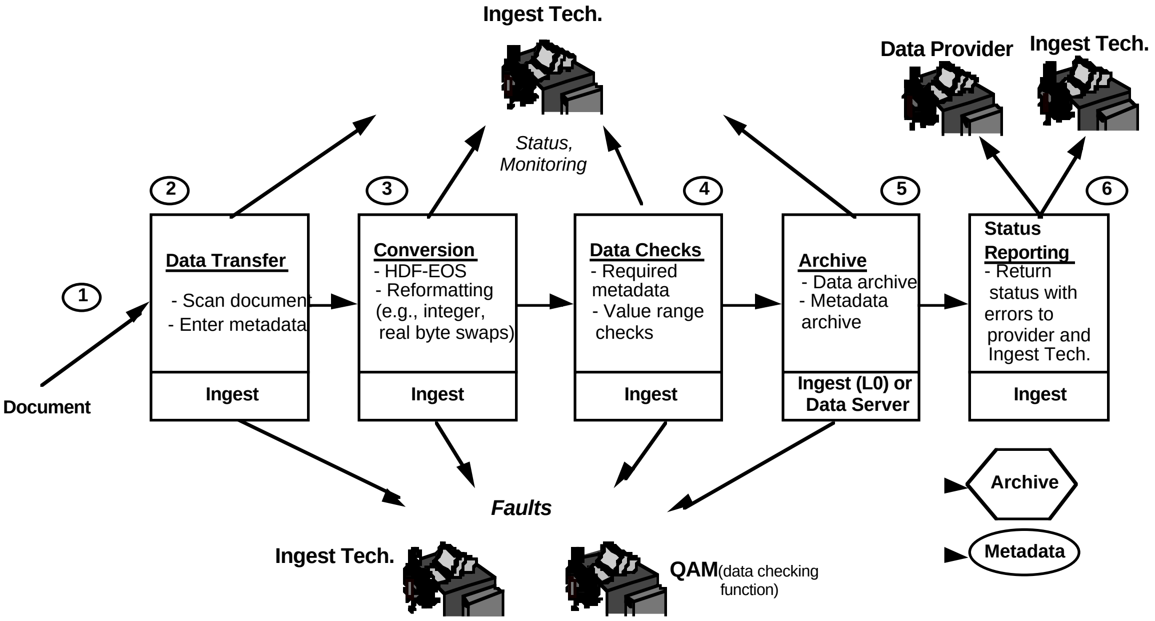


Figure 4.2.1.8-1. Document Ingest Scenario

Table 4.2.1.8-1 Document Ingest Scenario

Purpose and Precondition:

The operator-related precondition during normal operations for this scenario deals with the physical handling and checking of documents that must be performed prior to the start of document ingest.

Step	Operator/User	System	Purpose
1	The Ingest Technician compares the received document against the document ingest readiness checklist to ensure that the document and metadata are complete. The Ingest Technician then invokes the ingest client s/w via the GUI.		A document is received at the DAAC and is checked for readiness to ingest.
2	The Ingest Technician may monitor the status display showing subsequent ingest request processing and suspend, resume, and cancel requests. The Ingest Technician receives the device ID and inserts the document into the scanner when prompted to do so.	A device allocation is requested from the Data Server peripheral pool. The Ingest Technician is prompted to insert the document into the scanner. The digitized data is transferred from the scanner to working storage. Status of all involved devices is written to the event log. Peripherals are deallocated when data transfer is complete.	Transfer data from media to ECS Ingest.
3		Perform data conversion or reformatting as required for the particular data being ingested.	Convert data to ECS-supported format.
4		Ingest technician enters appropriate metadata parameters. Status of metadata parameter check is written to event log.	Insert of select metadata parameters.
5		Ingest Technician generates data server insert request to store data and metadata in the appropriate data server. Subscription (if any) is triggered to indicate availability of data when the archive process is completed.	Insert of data in the appropriate data server.
6	The Ingest Technician may view the history log. The request ID associated with this ingest process drops off the ingest status display at this time.	Email notification is sent to the data provider if a network address is available. Status messages are provided to the Ingest Technician and Ingest History Log when archiving is complete.	Completion of ingest session.

4.2.1.9 Document Modification Scenario

This scenario assumes that the updates to the multi-part document involves actual content changes that are made by the scientist. Those changes could involve modification to the narrative text and/or to html embedded in the document. Additionally, the document metadata, including keywords, may also be updated. Multiple-part documents are ingested and updated in the same way single-part documents are in ECS, because each part is processed separately at the DDSRV level. Each document in ECS has a "master document" that the DDSRV creates when a document is inserted. The master document tracks all of the document parts. For a single-part document, it simply identifies the single part. When a scientist describes a document part on the Ingest Client html forms, he/she indicates the document name to which the part belongs and identifies what part of the document it is, i.e, part 2. Document parts may be submitted separately (or together). They may also be submitted out-of-sequence, i.e., part 2 may be submitted before part 1.

The scenario begins when scientist decides that a document needs to be modified. First, the scientist needs to retrieve the parts of the document that need to be updated and have access to the document parts outside of ECS. Secondly, the appropriate text editor tool (e.g., WordPerfect, Word, Interleaf, etc.) is used outside the context of ECS to make the changes. For example, Word would be used to make changes to a document in Word format. Changes could be made to one or more parts of the document and/or new parts to the document could be added. Thirdly, the updated (or new) document parts need to be ingested and added to ECS.

There are several mechanisms in ECS through which the scientist can request the parts of the document to be updated. The figures below illustrate the http document request method where, via the ECS client, the scientist can request the document parts (individually). Each part is separately FTP'd back to the scientist at a designated file system location. Alternatively, the scientist could request either the entire document (or just the parts that need to be updated) through the standard distribution request mechanism.

Once the scientist completes the document updates outside of ECS and determines any relevant metadata changes, the ECS ingest processing can begin. Figure 4.2.1.9-1 and Figure 4.2.1.9-2 show how either a new or updated document part is ingested into ECS. To start the process, the scientist clicks on the Ingest icon to invoke the Ingest Subsystem Client. Ingest requires that two steps be performed in order to complete the ingest. First, the scientist fills in the relevant document metadata information on a series of html forms. In addition, the updated document parts are FTP'd by the scientist to a common (i.e., known) ECS location where the Ingest Subsystem can access them.

Once the scientist has completed providing the updated document and metadata, the Ingest Subsystem extracts the metadata information, performs range-checking where necessary on updated data and generates a new keyword list based on the ECS keyword master list.

When Ingest completes processing, it hands the metadata file and the document part(s) to the Document Data Server. If the document part received by the DDSRV is new, DDSRV creates a new document master and adds the part reference to the master. If the document part received is new, but the document master exists, it adds the part reference to the document master. For an existing document part that is being updated, the document master remains unchanged. The new document part is then inserted into the Document Repository.

DDSRV continues its processing with several more steps. Full-text search indexing is performed on the updated part and the indices are added to the Full-Text Search Index Repository. If metadata updates came with the document update, the Document Data Server deletes the existing metadata and replaces it with the new metadata. Note that this assumes that the metadata update is a complete replacement of the existing metadata. It should also be noted that the DDSRV does not validate external hypertext links. If a link goes away, the users must submit a trouble ticket to report the problem. Hypertext links can be updated through the Illustra Web GUI by the operator. After all of the DDSRV updates are completed, DDSRV returns status information back to the Ingest Subsystem that is fed back to the user through the ECS Client and the scenario is completed.

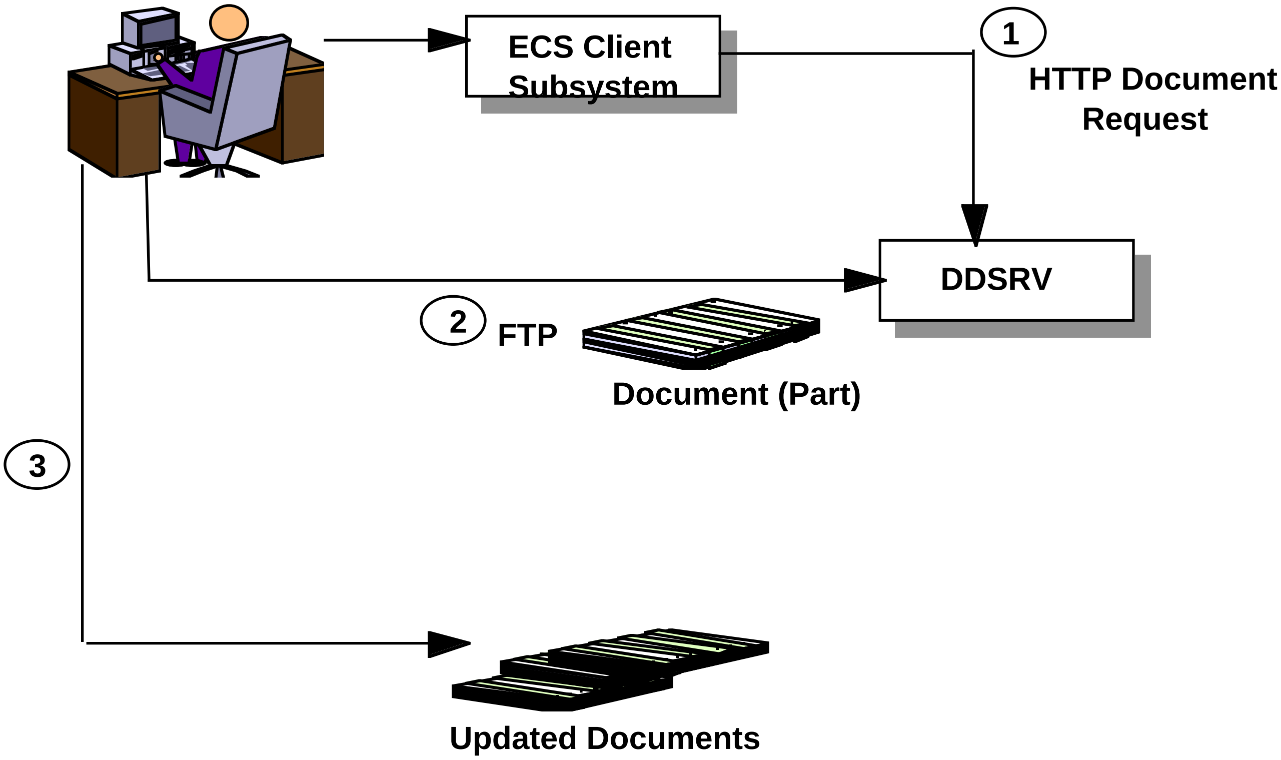


Figure 4.2.1.9-1 Retrieving an ECS Document for Update

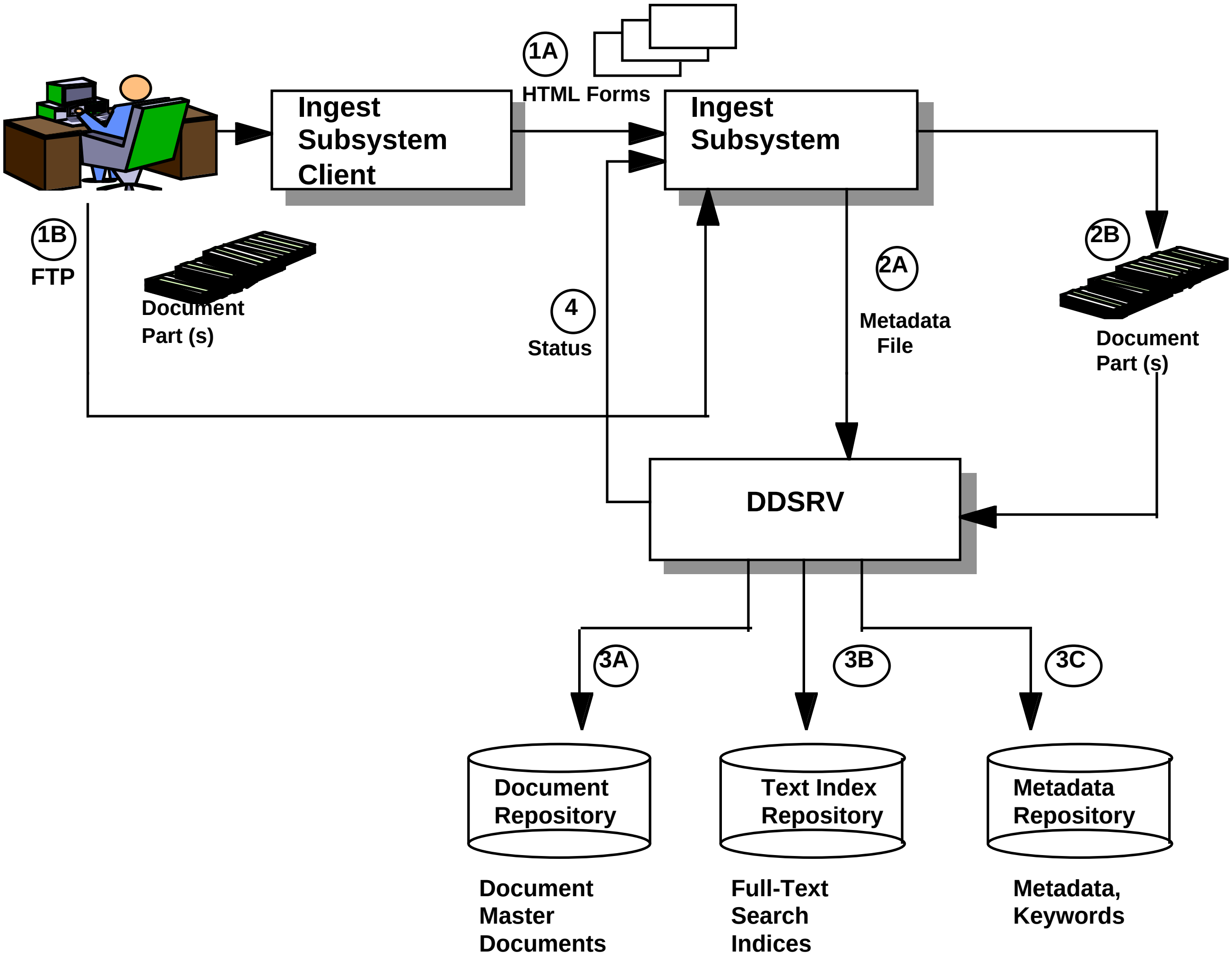


Figure 4.2.1.9-2 Updating a Multi-Part Document

Table 4.2.1.9-1 Retrieving an ECS Document for Update (1 of 2)

Step	Operator/User	System	Purpose
1	User selects to receive an ECS stored document or a single part of a multi-part ECS document.	ECS Client Subsystem provides user interface and sends the http document request to the DDSRV.	To get a document (or one of its parts) from the ECS Document Repository.
2		The DDSRV FTPs the requested document (or part) to the desired file system location.	To place the requested document in a file system location that the scientist can access.
3	User employs appropriate text editor tool to update ECS document (or part).		To make the desired document native format update outside ECS.

Table 4.2.1.9-2 Updating a Multi-Part Document (1 of 2)

Step	Operator/User	System	Purpose
1A	User specifies document metadata for update.	Ingest Subsystem Client displays series of html forms that define the document metadata to be updated.	To accept the document metadata for update in ECS.
1B	User FTPs document part(s) to common (known) ECS location.	Ingest Subsystem retrieves the updated document part(s) for processing.	To accept the updated document parts into ECS.
2A		Ingest Subsystem performs necessary document metadata validation and keyword list development. Then, Ingest sends the newly created metadata file to the DDSRV for processing.	To validate the document metadata.
2B		Ingest sends the document part(s) to DDSRV for processing.	To position the documents for insertion into the Document Repository.
3A		DDSRV updates the master document if necessary and inserts each document part (one-at-a time) into the Document Repository.	To make the updated document part(s) available through ECS.

Table 4.2.1.9-2 Updating a Multi-Part Document (2 of 2)

Step	Operator/User	System	Purpose
3B		DDSRV creates full text search indices on the updated document and places them into the Full-Text Search Index Repository.	To permit full-text searches on the updated document through ECS.
3C		DDSRV inserts the updated metadata and keyword lists into the Metadata Repository.	To make the updated document metadata available through ECS.
4		DDSRV returns the status of the document processing back to the Ingest Subsystem for display through the ECS Client.	Notifies the user that the document update was completed or that it failed.

4.2.2 Science Data Archival Activities

Science data archival activities at ECS DAACs include:

- data insertion
- data checksum calculation
- data storage in the permanent repository
- subscription processing
- data retrieval to working storage for transfer to either the processing or distribution systems

Subsetting and subsampling of selected data products is also described based on capabilities and functions described in DID 304 Appendix F. The objective of the following paragraphs is to demonstrate that the working storage and data archival process is largely automated, however the DAAC operations staff is required to support data archive administration operations, resolve problems, periodically monitor working storage and archival operations, and coordinate with the appropriate external/internal sources to resolve resource schedule conflicts.

One topic that needs clarification is the idea of queuing requests. In the following scenarios, the term "queue" is used. This brings to mind a First In First Out (FIFO) data structure. With object oriented design and process threads (p-threads) the logical structure used is provided by OO-DCE and it is termed a vector. The vector is similar to a queue but is not necessarily processed in a FIFO manner. Each p-thread within a vector has an associated priority based on the session initiating the threads and some other data. Items in the queue are processed according to one of several paradigms available within OO-DCE. Some threads may be processed before other threads and some threads may receive more overall CPU cycles in a time sharing environment. (e.g. six request threads are in the vector. Four are priority 6 (10 is the highest) two are priority 2. In a time sharing service paradigm, the four priority 6 requests may each get 20% of the available CPU cycles with each priority 2 request receiving 10%). This is a gross example but it hopefully illustrates a possible paradigm. When the term "queue" is used below it should be interpreted as a vector in the manner described above.

4.2.2.1 Data Insertion Scenario (nominal)

This scenario describes the insertion of data into a Data Server at an ECS DAAC. This process is largely automated with validation errors being manually processed by the QA staff. Data and associated metadata can be received from numerous sources including: the Ingest Subsystem, the Processing Subsystem, other DAACs, and Users. This scenario will focus on a data insert from the Processing Subsystem. The validation, insertion, and subscription processing process is described in Table 4.2.2.1-1. A graphical representation of this scenario is depicted in Figure 4.2.2.1-1.

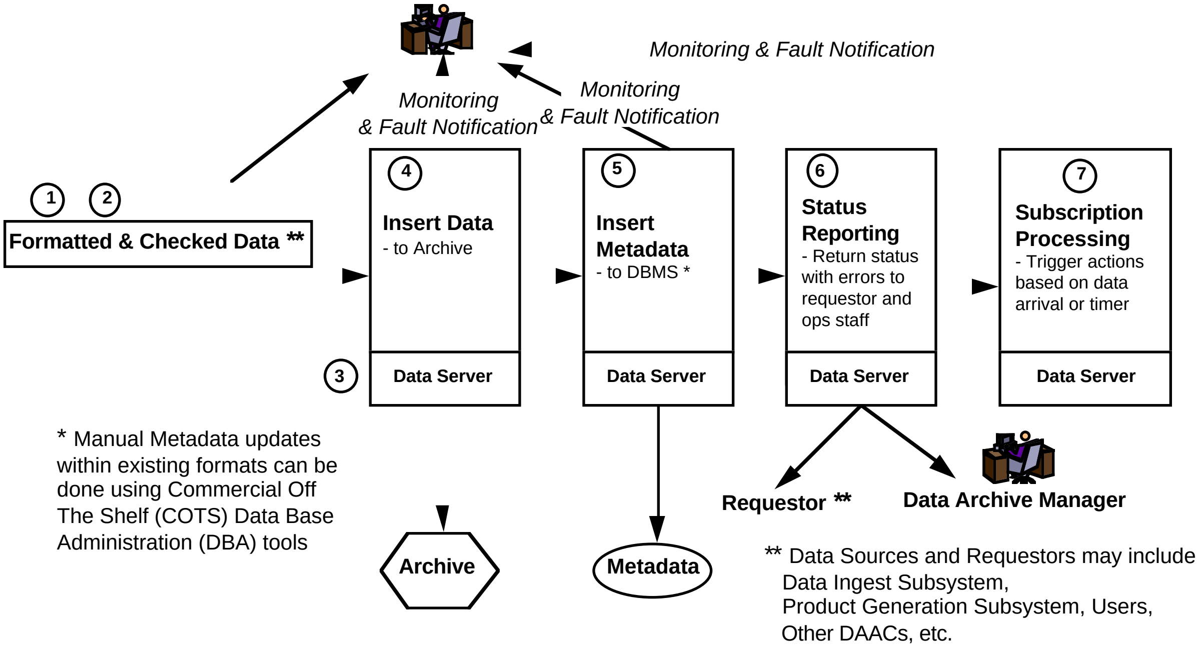

Monitoring & Fault Notification
Monitoring
& Fault Notification
Monitoring
& Fault Notification
1
2
Formatted & Checked Data **
4
Insert Data
- to Archive
5
Insert Metadata
- to DBMS *
6
Status Reporting
- Return status with errors to requestor and ops staff
7
Subscription Processing
- Trigger actions based on data arrival or timer
3
Data Server
Data Server
Data Server
Data Server
* Manual Metadata updates within existing formats can be done using Commercial Off The Shelf (COTS) Data Base Administration (DBA) tools
Archive
Metadata
Requestor **
Data Archive Manager
** Data Sources and Requestors may include Data Ingest Subsystem, Product Generation Subsystem, Users, Other DAACs, etc.

Figure 4.2.2.1-1. Data Insertion Scenario (nominal)

Table 4.2.2.1-1. Data Insertion Scenario (nominal) (1 of 2)

Purpose and Precondition:

The Data Server is required to receive, validate, and insert data and metadata into the archive received from authorized users. An authorized user (in this case, the Processing Subsystem) initiates a data transfer session by sending a Data Insert Request.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Data Server Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files provided by MSS.	The Processing Subsystem sends a Data Insert Request to the Science Data Server. Receipt of the Request is logged, and a request identifier is associated with the Data Insert Request. The content of the request is validated. Validation failure results in a rejection message. Validation success results in the request being queued for later processing.	Initiate session between the Processing Subsystem and a Data Server.
2	The Operator may check request status at any time.	The queued Data Insert Request is reached and processing begins. Associated data granules and metadata are transferred from the Processing Subsystem to the Data Server working storage. Data transfer status (including recoverable errors) are indicated in the event log.	Transfer data from a Processing Subsystem to a Data Server.
3		The metadata update file(s) produced by the associated data product PGEs are validated for completeness and correctness. Validation success or failure is logged with the associated Data Insert Request Identifier and the appropriate status message is returned to the Processing Subsystem.	Validate metadata received from the Processing Subsystem.
4		Upon successful validation of the metadata update file, Science Data Server sends a Data Storage Request to Storage Management. The data granules in working storage associated with the Data Storage Request are stored. The Archive Activity Log records each data product being stored and storage status of each storage operation. A checksum value is calculated for each data object associated with each granule. The checksum value, storage status, and other selected metadata is forwarded to the Science Data Server in a status message upon completion of the Data Storage Request.	Store data granules in the permanent archive.

Table 4.2.2.1-1. Data Insertion Scenario (nominal) (2 of 2)

Step	Operator/User	System	Purpose
5		Science Data Server receives and logs the Data Storage Request status message from Storage Management. The additional metadata items are validated. The PGE produced metadata update file and the storage management provided metadata are loaded into the metadata database. The status of the metadata load is entered in the event log.	Store metadata.
6		The Science Data Server logs completion of the Data Insert Request in the event log and reports completion of the Data Insert Request to the Data Archive Manager, the operator console and to the insert Requester (the Processing Subsystem in this case). Each of the above entities would also be notified if the request failed and reason(s) for failure is/are identified.	Report Data Insert Request Status.
7		The Science Data Server will then examine the event list for all subscriptions for that event. Subscription notifications are sent to the appropriate entities as appropriate and distribution processing is initiated. The Science Data Server sends an Advertisement Update Message to the Advertisement Server to advertise the new data.	Process subscriptions based on newly inserted data.

4.2.2.2 Data Insertion Scenario (fault)

This scenario describes the insertion of data into a Data Server at an ECS DAAC and the sequence of events when a fault occurs. This process is largely automated with validation errors being manually processed by the QA staff. Data and associated metadata can be received from numerous sources including: the Ingest Subsystem, the Processing Subsystem, other DAACs, and Users. This scenario will focus on a data insertion and fault processing from the Processing Subsystem. The validation, insertion, and subscription fault processing procedures are described in Table 4.2.2.2-1. A graphical representation of this scenario is depicted in Figure 4.2.2.2-1.

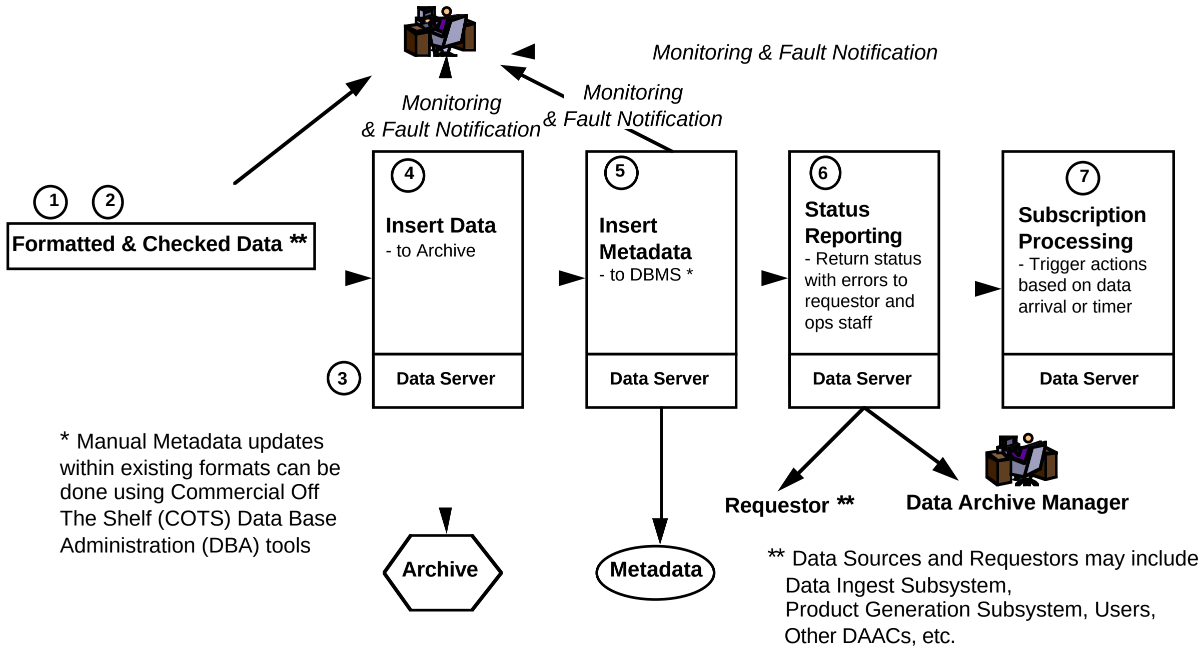

Monitoring & Fault Notification
Monitoring & Fault Notification
Monitoring & Fault Notification
1
2
Formatted & Checked Data **
4
Insert Data
- to Archive
5
Insert Metadata
- to DBMS *
6
Status Reporting
- Return status with errors to requestor and ops staff
7
Subscription Processing
- Trigger actions based on data arrival or timer
3
Data Server
Data Server
Data Server
Data Server
* Manual Metadata updates within existing formats can be done using Commercial Off The Shelf (COTS) Data Base Administration (DBA) tools
Archive
Metadata
Requestor **
Data Archive Manager
** Data Sources and Requestors may include Data Ingest Subsystem, Product Generation Subsystem, Users, Other DAACs, etc.

Figure 4.2.2.2-1. Data Insertion Scenario (fault)

Table 4.2.2.2-1. Data Insertion Scenario (fault) (1 of 3)

Purpose and Precondition:

The Data Server is required to receive, validate, and insert data and metadata into the archive received from authorized users. An authorized user (in this case, the Processing Subsystem) initiates a data transfer session by sending a Data Insert Request.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Data Server Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files provided by MSS.	The Processing Subsystem sends a Data Insert Request to the Science Data Server. Receipt of the Request is logged, and a request identifier is associated with the Data Insert Request. The content of the request is validated. Validation failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message and the Data Insert Request are forwarded to QA for evaluation and correction. The Data Insert Request is closed and the Data Insert Session is terminated. Validation success results in the request being queued for later processing.	Initiate session between the Processing Subsystem and a Data Server.
2	The Operator may check request status at any time.	The queued Data Insert Request is reached and processing begins. Associated data granules and metadata are transferred from the Processing Subsystem to the Data Server working storage. Data transfer status (including recoverable errors) are indicated in the event log.	Transfer data from a Processing Subsystem to a Data Server.

Table 4.2.2.2-1. Data Insertion Scenario (fault) (2 of 3)

Step	Operator/User	System	Purpose
3		The metadata update file(s) produced by the associated data product PGEs are validated for completeness and correctness. Validation failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message and the metadata update file(s) are forwarded to QA for evaluation and correction. The Data Insert Request is closed and the Data Insert Session is terminated.	Validate metadata received from the Processing Subsystem.
4		Upon successful validation of the metadata update file, Science Data Server sends a Data Storage Request to Storage Management. The data granules in working storage associated with the Data Storage Request are stored. The Archive Activity Log records each data product being stored and storage status of each storage operation. Storage failure results in a rejection message being sent to the requester. The rejection message is entered in the archive activity log and forwarded to the Data Archive Manager and the operator for corrective action. (The most likely reason for a storage failure is a physical hardware problem.) The Data Storage Request continues or is discarded in accordance with DAAC Policy. The checksum value, storage status, and other selected metadata is forwarded to the Science Data Server in a status message upon completion of the Data Storage Request.	Store data granules in the permanent archive.
5		Science Data Server receives and logs the Data Storage Request status message from Storage Management. Validation failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message and the Data Storage Request Status Message are forwarded to QA for evaluation and correction. The Data Insert Request continues or is discarded in accordance with DAAC Policy. The PGE produced metadata update file and the storage management provided metadata are loaded into the metadata database. The status of the metadata load is entered in the event log. Load failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message and the Data Storage Request Status Message are forwarded to QA for evaluation and correction. The Data Insert Request continues or is discarded in accordance with DAAC Policy.	Store metadata.

Table 4.2.2.2-1. Data Insertion Scenario (fault) (3 of 3)

Step	Operator/User	System	Purpose
6		The Science Data Server logs completion of the Data Insert Request in the event log and reports completion of the Data Insert Request to the Data Archive Manager, the operator console and to the insert Requester (the Processing Subsystem in this case). Each of the above entities would also be notified if the request failed and reason(s) for failure is/are identified.	Report Data Insert Request Status.
7		The Science Data Server will then examine the event list for all subscriptions for that event. Subscription notifications are sent to the appropriate entities as appropriate and distribution processing is initiated.	Process subscriptions based on newly inserted data.

4.2.2.3 Data Archive Configuration Maintenance Scenario - Media Refresh

This scenario describes the process of media refresh within the archive. In Release A this is predominantly a manual process. Release B is more automated. This scenario (described in Table 4.2.2.3-1) will focus on physical media refresh of an archive volume due to excessive wear/utilization or manufacturing/quality problems which result in a number of correctable bit errors on a volume. A graphical representation of this scenario is depicted in Figure 4.2.2.3-1.

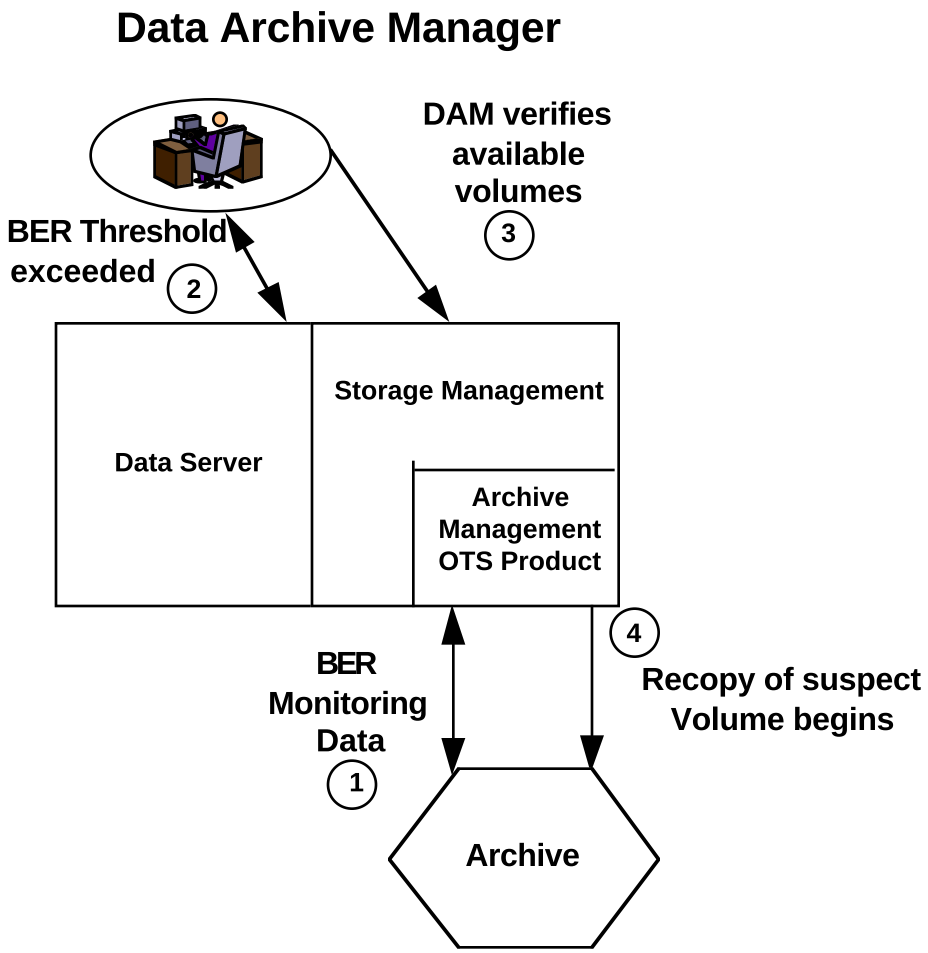


Figure 4.2.2.3-1. Data Archive Configuration Maintenance - Media Refresh - Scenario

Table 4.2.2.3-1. Data Archive Configuration Maintenance - Media Refresh - Scenario

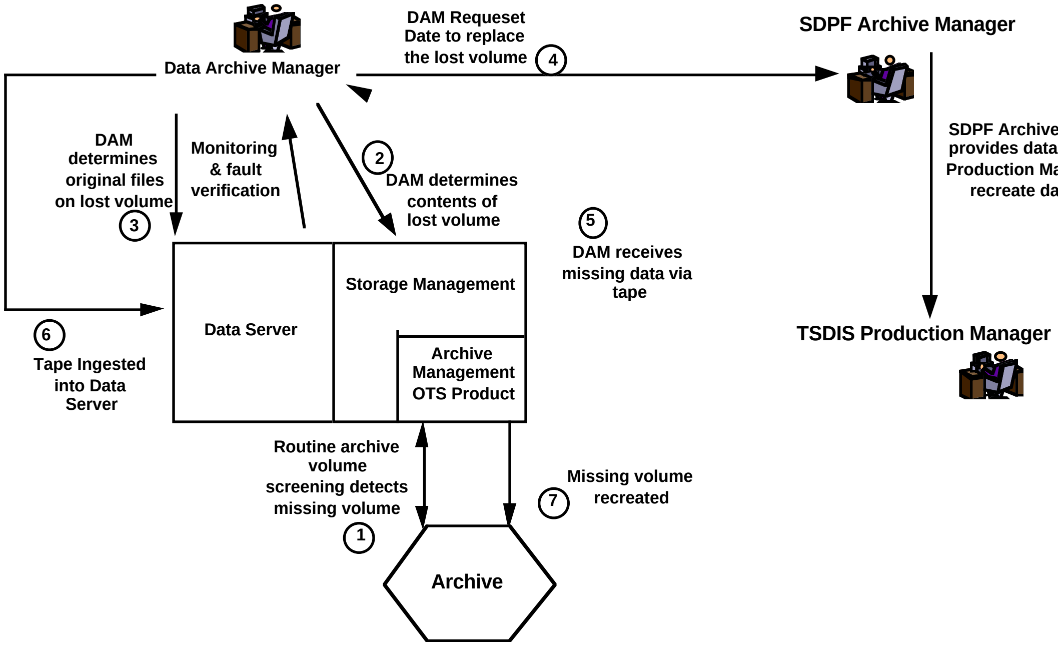
Purpose and Precondition:

This scenario describes one several mechanisms that will insure long term viability of the archived data. The scenario assumes the following: (1) The archive consists of hardware devices that report the number of Bit Errors and Corrections detected on each volume. (2) The Archive Management OTS product can receive and record Bit Errors and Corrections reported by the hardware. (3) The Archive Management OTS product can take corrective action based on the reported Bit Error Rate (BER). (4) A Unix Cron Tab entry has been made by the Data Archive Manager that specifies when media refresh operations should begin according to DAAC Policy.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Data Server Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files provided by MSS.	During a Read or Write operation a hardware device reports one or more correctable Bit Errors to Archive Management OTS Product.	Routine viability monitoring of archive data volumes.
2		The Archive Management OTS Product updates the corresponding entry in its volume database with the new error information and compares the total to the DAAC Policy defined error threshold. If Correctable Bit Errors exceed the established threshold. An alarm is recorded in the Archive Activity Log and a copy is sent to the Data Archive Manager and the operator console. The archive management OTS product updates the corresponding entry in the volume database as Read-Only and BER-Limit Exceeded. This volume database attribute update is recorded in the Archive Activity Log.	Update and test the number of correctable bit errors against the BER threshold established by DAAC Policy.
3	The Data Archive Manager or operator verifies that sufficient blank volumes exist in the appropriate media volume group to support media refresh operations.		Verify media is available for refreshing. (Optional, insufficient media will result in different alarms and requests for corrective action).
4		The Unix CRON Tab entry for media recopying on the Archive Management OTS Product begins at the appointed time. The OTS volume database is searched for BER-Limit Exceeded equal to 'yes'. Each faulty volume is recopied to a blank volume. The faulty volume is ejected. The OTS volume database is updated with the new volume's information. The faulty volume's information is marked for deletion. Initiation and completion of each recopy operation is recorded in the Archive Activity Log. The faulty volume(s) is(are) disposed of according to DAAC Policy.	Recopy and eject the suspect medium.

4.2.2.4 Data Archive Configuration Maintenance Scenario - Lost Volume

This scenario describes the process of media recovery as a result of theft, hardware failure, or some other catastrophic event resulting in the destruction or loss of one or more archive data volumes. The Lost Volume recovery process is described in Table 4.2.2.4-1. A graphical representation of this scenario is depicted in Figure 4.2.2.4-1.



Data Archive Manager
DAM Requeset Date to replace the lost volume
4
SDPF Archive Manager
SDPF Archive Manager provides data to TSIDS Production Manager to recreate data lost
DAM determines original files on lost volume
3
Monitoring & fault verification
2
DAM determines contents of lost volume
5
DAM receives missing data via tape
Storage Management
Data Server
Archive Management OTS Product
TSDIS Production Manager
6
Tape Ingested into Data Server
Routine archive volume screening detects missing volume
1
Missing volume recreated
7
Archive

Figure 4.2.2.4-1 Data Archive Configuration Maintenance - Lost Volume - Scenario

Table 4.2.2.4-1 Data Archive Configuration Maintenance - Lost Volume - Scenario

Purpose and Precondition:

This scenario describes the available recovery procedures for a lost or destroyed media volume. This scenario focuses on recovering data from SDPF or EDOS. A similar scenario would be used for other data providers. The scenario assumes the following: (1) EDOS and SDPF maintain all L0 data received from instruments. (2) There is no direct interface or API between ECS and EDOS or SDPF to request data from there L0 archives. (3) For the purpose of the textual description, the scenario will assume the missing volume was located in 4Q 1998 and contained TMI L1A data.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Data Server Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files provided by MSS.	During a routine screening of the archive volumes, the Archive Management OTS Product discovers a missing volume. The OTS volume database is queried to see if the volume was exported to shelf storage or another site. The database query returns a volume should be present status. An alert is recorded in the Archive Activity Log and a copy is sent to the Data Archive Manager (DAM) and the operator console.	Routine archive volume screening detects a missing volume.
2	The DAM queries the Archive Management OTS Product to obtain hard copy of the missing volume's header information and the ECS name of each file on the volume.	The Archive Management OTS Product searches its volume database to retrieve information on the missing volume.	Determine the contents of the missing volume from an ECS perspective.
3	The DAM queries the Science Data Server's Metadata database using the ECS name of each file from the lost volume to obtain the original ingest name of each file.	The Science Data Server queries the metadata database to identify the original name attribute associated with the provided ECS name.	Determine the associated ingest name for each missing file to facilitate reordering.
4	The DAM call the SDPF to describe the missing data. Arrangements are made to fax or email the list of requisite files to the SDPF operator for retrieval and processing.		Request the missing data files from the appropriate L0 archive.
5	The DAM receives a tape volume (in an ECS approved format) containing the requisite missing files from TSDIS.		Missing files are processed and distributed to the appropriate DAAC.
6	The DAM schedules the volume for ingestion into the DAAC's archive.		Missing volume data re-ingested.
7		Data is ingested and placed in working storage. Storage Management recreates the missing data volume.	Missing archive volume is recreated.

4.2.2.5 Data Type Service Modification Scenario

This scenario describes the process for adding or modifying services related to a specific Earth Science Data Type (ESDT). The Data Server is a framework that provides the infrastructure (i.e. persistent storage, queues resource management, advertisement coordination, etc.) in which ESDT's and their services can execute. This process is described in Table 4.2.2.5-1. A graphical representation of this scenario is depicted in Figure 4.2.2.5-1.

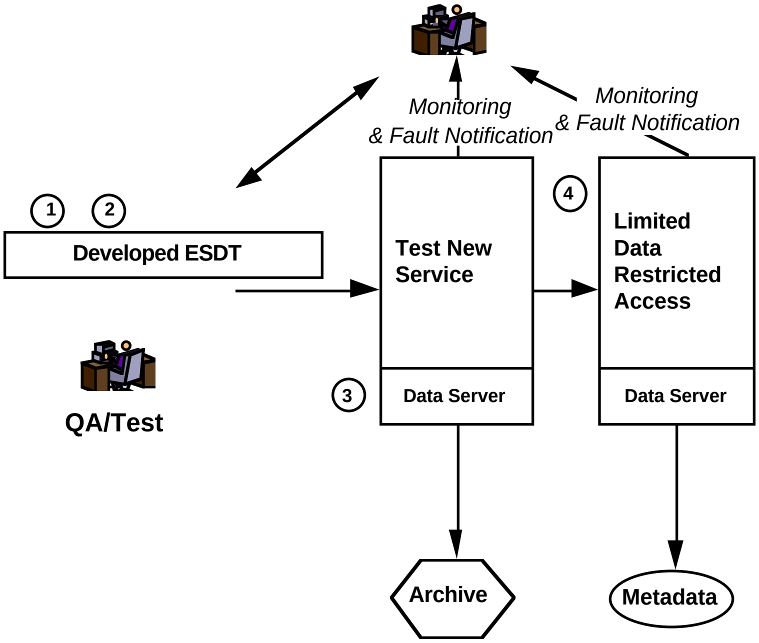


Figure 4.2.2.5-1. Data Type Service Modification Scenario

Table 4.2.2.5-1. Data Type Service Modification Scenario

Purpose and Precondition:

This scenario describes the steps necessary modify and test an ESDT service. The scenario assumes the following: (1) the basic Data Server services such as Insert, Acquire, Browse, etc., have been fully tested. (2) A fully verified Data Base and Archive population exists.

Step	Operator/User	System	Purpose
1	The development staff would use local resources to code and test the modified ESDT service, using test data stored locally on the development station and lightweight test drivers/shells to invoke the service.		Develop and perform initial testing of the modified service.
2	The development personnel would integrate the new service into a Data Server framework. A "test" Data Server instantiation is then created that includes the existing ESDT and the new service.		Create a new testing framework.
3	Development and test personnel evaluate the modified service.	The "test" data server makes use of private advertisements to limit access to development and test personnel. Read-Only access to operational data and metadata is used if appropriate.	Perform Operational Testing and Evaluation of the new service.
4	Development and test personnel are satisfied the modified service is performing properly. A Data Server instantiation is then created that includes the existing ESDT and the new service. Configuration Management data is updated.		Modified service made available to the community.

4.2.2.6 Bad Data Scenario

This scenario touches on several ECS issues (e.g. who determines what is considered bad, how do quality personnel differentiate between bad or dubious ancillary data versus bad or dubious parent products in a product production string, who resolves the issues of "bad to one investigator is not necessarily bad to another?", etc.) Quality flags are present in the database model to differentiate between PGE produced quality, DAAC quality assessments and science quality assessments. One potential Bad Data Scenario is described in Table 4.2.2.6-1. A graphical representation of this scenario is depicted in Figure 4.2.2.6-1.

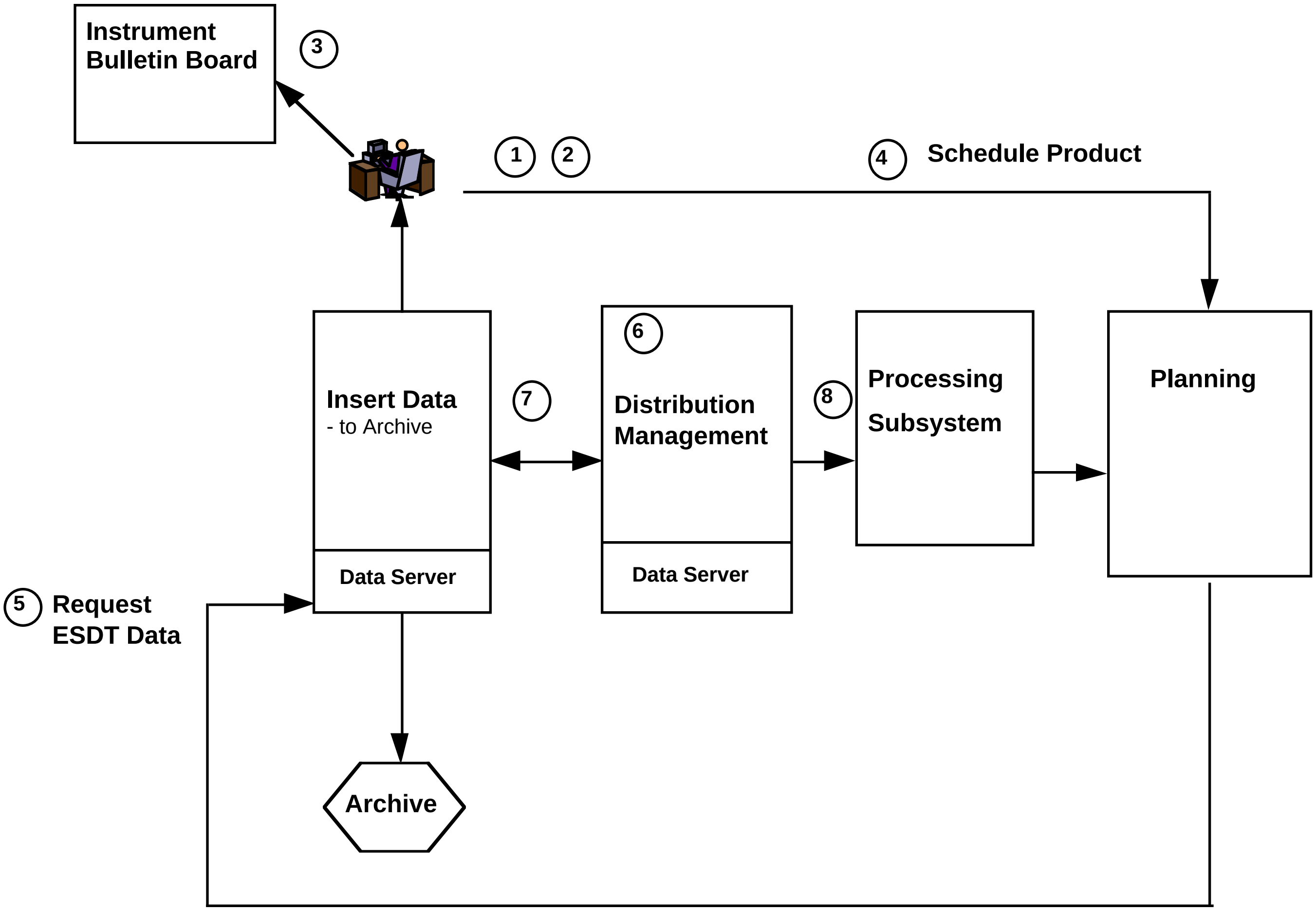


Figure 4.2.2.6-1. Bad Data Scenario

Table 4.2.2.6-1. Bad Data Scenario (1 of 2)

Purpose and Precondition:

Since issues of quality determination of PGE output have yet to be answered, this scenario will focus on the improper calibration of a PGE. Improper calibration is identified after three months using a forum of DAAC quality personnel, instrument team members, and science users. Some volume of data has been produced and distributed prior to discovery of this problem. Due to the anticipated loads of routine and ad hoc production as well as the cost constraints imposed on the implementation, it will not be possible for and ECS DAAC to reprocess and redistribute every descendant of the improperly calibrated PGE. The PGE calibration will be corrected and the output products produced by the PGE will be re-produced. Another gray area needing definition and refinement is the area of reprocessing and what constitutes a new version of data. This scenario will avoid this issue. Once a revised product has been produced. A message will be placed on the instrument bulletin board identifying the nature of the faulty calibration and the affected data products. The onus is on the investigator to periodically check instrument boards on interest and to reorder revised products as appropriate.

Step	Operator/User	System	Purpose
1	DAAC quality personnel, instrument team members, and science users meet to identify suspected problems with a PGE.		Questionable output from a PGE traced to improper calibration.
2	Problem is identified as calibration related. PGE is recalibrated and tested.		DAAC personnel and instrument team members correct and test the PGE.
3	The operator places a notice on the instrument bulletin board identifying the problem and the affected data.		Notice placed on the instrument bulletin board.
4	The operator creates a data processing request and sends this data to planning.		Operator request reprocessing of PGE descendants.
5		Planning reaches the data processing request and establishes a session with Science Data Server to build an ESDT data acquisition request. The requisite data is identified and an acquire via ftp push request is generated.	Identify the input data to produce the requested output from the PGE.
6		Distribution Management logs the acquire via ftp push request. When the request thread is processed, Distribution Management sends a Data Retrieval Request to Storage Management listing the granules of high interest to be retrieved.	Request requisite data collection from the archive.

Table 4.2.2.6-1. Bad Data Scenario (2 of 2)

Step	Operator/User	System	Purpose
7		Storage management logs and queues the Data Retrieval Request. When the request is reached in the request queue, Storage Management requests the appropriate granules be retrieved from the archive via the Archive Management OTS Product. The granules are placed on the Working Storage and a Data Retrieval Request completed message is logged and sent to Distribution Management.	Retrieve the appropriate granules and place them on the user pull volume.
8		Distribution Management utilizing login, system, and security information in the Acquire via ftp Push, pushes the high interest granules to the Processing Subsystem. Distribution Management logs a distribution complete message and sends a distribution completed notification to the Processing Subsystem.	Distribute data and notify Processing that data is available.
9		The Processing Subsystem will process the appropriate data and then data will be inserted into the Data Server Archive via 4.2.2.1 Data Insert Scenario.	Processing and storage of data.

4.2.2.7 Data Server Startup/Shutdown

The startup and shutdown scenario relies on a command from the MSS supplied SNMP agent to load and start each subsystem host's Unix OS. The agent insures each host has reached a quiescent state, and then the subsystem startup scripts are initiated. The Metadata CSC (Sybase in Release A) starts first followed by Data Server Subsystem Overall System Management (DSS-OSM). The operator initiates a subsystem startup which triggers startup scripts associated with each Subsystem software component. At this point, Storage Management begins first to insure availability and accessibility of the physical devices. Distribution Management then starts and insures it has access to the available physical devices. Science Data Server starts and insure communication to internal subsystem and external entities. Post startup equipment status is collected and forwarded to MSS. For shutdown, new requests and searches are suspended. Database updates are completed, and sessions are checkpointed, disconnected and stored. The startup process is described in Table 4.2.2.7-1. The shutdown process is described in Table 4.2.2.7-2. A startup from abnormal shutdown process is described in Table 4.2.2.7-3.

Table 4.2.2.7-1. Data Server Startup/Shutdown Scenario

Purpose and Precondition:

This scenario will focus on subsystem startup from a quiescent Unix OS.

Step	Operator/User	System	Purpose
1		MSS host startup sent via SNMP Agent. All hardware powered up.	SNMP Agent powers up DSS hardware and boots Unix OS as appropriate.
2	Operators may login to the Unix Host directly and view *errlog* and other Unix logs at any time.	Software startup sent via SNMP to Data Server hosts.	MSS initiates the Data Server Startup sequence.
3	Operator signs on via the *DSS System Login Dialog* Screen and is transferred to the *System State* Screen. The operator clicks *Startup*.	The COTS Metadata Management product is started along with the DSS Overall System Management.	Initialize COTS product. (Some Data Server persistence data may be stored in the metadata management product). Start DSS-OSM.
4		The AMASS COTS product is started.	AMASS software starts, performs internal consistency checks and inventories the archive robotics.
5	The operator will see the state of each CSCI or COTS product change from *Down* to *Active* on the *System State* Screen when component initialization is completed.	Standalone StorageResourceManagement processes are started. One for each resource type. (network, staging disk, tape, CD ROM) StagingMonitor and PullMonitor started.	Storage Management start and check physical devices after metadata COTS product finishes.
6	The operator may examine the progress of *Down* CSCI by pulling down the *Overall Systems Screens* option and selecting *Logs & Reports (MSS).*	Web-based software started on the DDSRV host.	Document Data Server Starts.
7		DistributionServer process is started in standalone mode.	Data Distribution starts.
8		ScienceDataServer and SubscriptionServer processes are started in standalone mode.	Data Server Starts.
9	At this point the operator will see the state of all CSCIs and COTS products is *Active*.	A status message is provided to MSS detailing available and unavailable equipment via the MSS API.	Report subsystem readiness to MSS.

Table 4.2.2.7-2. Data Server Operator Shutdown Scenario

Purpose and Precondition:

This scenario will focus on subsystem shutdown by the operator.

Step	Operator/User	System	Purpose
1	Operator signs on via the *DSS System Login Dialog* Screen and is transferred to the *System State* Screen. The operator clicks *Shutdown*.	ScienceDataServer and SubscriptionServer processes are stopped.	Data Server Stops. Requests and searches are suspended. Database updates are completed, sessions checkpointed, disconnected and stored.
2		DistributionServer process is stopped.	Data Distribution stops.
3	The operator will see the state of each CSCI or COTS product change from *Active* to *Down* on the *System State* Screen when component shutdown is completed.	Web-based software stopped on the DDSRV host.	Document Data Server Shuts down.
4	The operator may examine the progress of *Down* CSCI by pulling down the *Overall Systems Screens* option and selecting *Logs & Reports (MSS).*	Standalone StorageResourceManagement processes are stopped. One for each resource type. (network, staging disk, tape, CD ROM) StagingMonitor and PullMonitor stopped.	Storage Management check physical devices and stop after metadata COTS product finishes.
5		The AMASS COTS product is stopped.	The archive robotics shut down and AMASS software stops.
6		The COTS Metadata Management product is stopped along with the DSS Overall System Management.	Shutdown COTS product. (Some Data Server persistence data may be stored in the metadata management product). Stop DSS-OSM.
7	At this point the operator will see the state of all CSCIs and COTS products is *Down*.	A status message is provided to MSS detailing available and unavailable equipment via the MSS API.	Report subsystem readiness to MSS.
8	Operators may login to the Unix Host directly and view *errlog* and other Unix logs at any time.	Software shutdown sent via SNMP to Data Server hosts.	MSS initiates the Data Server Shutdown sequence.
9		MSS host shutdown sent via SNMP Agent. All hardware powered down.	SNMP Agent shuts down Unix OS as appropriate and powers down DSS hardware.

Table 4.2.2.7-3. Data Server Startup from Abnormal Shutdown Scenario

Purpose and Precondition:

This scenario will focus on subsystem startup after an abnormal shutdown.

Step	Operator/User	System	Purpose
1		MSS host startup sent via SNMP Agent. All hardware powered up.	SNMP Agent powers up DSS hardware and boots Unix OS as appropriate.
2	Operators may login to the Unix Host directly and view *errlog* and other Unix logs at any time.	Software startup sent via SNMP to Data Server hosts.	MSS initiates the Data Server Startup sequence.
3	Operator signs on via the *DSS System Login Dialog* Screen and is transferred to the *System State* Screen. The operator clicks *Startup*.	The COTS Metadata Management product is started along with the DSS Overall System Management.	Initialize COTS product. (Some Data Server persistence data may be stored in the metadata management product). Start DSS-OSM.
4		The AMASS COTS product is started.	AMASS software starts, performs internal consistency checks and inventories the archive robotics.
5	The operator will see the state of each CSCI or COTS product change from *Down* to *Active* on the *System State* Screen when component initialization is completed.	Standalone StorageResourceManagement processes are started. One for each resource type. (network, staging disk, tape, CD ROM) StagingMonitor and PullMonitor started.	Storage Management start and check physical devices after metadata COTS product finishes.
6	The operator may examine the progress of *Down* CSCI by pulling down the *Overall Systems Screens* option and selecting *Logs & Reports (MSS).*	Web-based software started on the DDSRV host.	Document Data Server Starts.
7		DistributionServer process is started in standalone mode.	Data Distribution starts.
8		ScienceDataServer and SubscriptionServer processes are started in standalone mode.	Data Server Starts.
9	At this point the operator will see the state of all CSCIs and COTS products is *Active*.	A status message is provided to MSS detailing available and unavailable equipment via the MSS API.	Report subsystem readiness to MSS.

4.2.2.8 ESDT Insert/Delete Scenario

Summary

This scenario begins at the point where the server session receives an Insert command request from the client. The client for this scenario is the Ingest subsystem. The session determines that the already established working collection is required to service the command. The working collection instantiates an 'empty' ESDT of the correct type using dynamic binding. Three main operations are then performed. First the science data file(s) are sent to the archive for storage, second the filenames and unique granule ID are added to the granule metadata and third the metadata is written to the database.

The unique granule ID is set during insertion of the granule metadata. The ID is passed back to the working collection as the result of a successful Externalize operation. Ultimately this ID will be returned to the client. Table 4.2.2.8-1 describes the Data Server's approach to processing ESDT Insert requests. The ESDT Delete request is described in table 4.2.2.8-2.

Table 4.2.2.8-1. ESDT Insert Scenario (1 of 2)

Purpose and Precondition:

The Data Server is required to archive science data file(s) and update and store the metadata for an ESDT.

The following preconditions are assumed for this scenario: ingest supplies a validated file containing P=V metadata values; ingest supplies a file containing scientific data; a work area has already been assigned; a connection has already been established to the dataserver; a working collection already exists; data has been staged ready for insertion into the database; the Insert request is as a result of call from the Ingest Subsystem; metadata stream has already been validated.

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Data Server Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files provided by MSS.	The Ingest Subsystem sends a ESDT Insert Request to the Science Data Server. Receipt of the Request is logged, and a request identifier is associated with the ESDT Insert Request. The content of the request is validated. Validation success results in the request being queued for later processing.	Initiate session between the Ingest Subsystem and a Data Server.

Table 4.2.2.8-1. ESDT Insert Scenario (2 of 2)

Step	Operator/User	System	Purpose
2	The Operator may check request status at any time.	Science Data Server sends a Data Storage Request to Storage Management. The data granules in working storage associated with the ESDT Insert Request are stored. The Archive Activity Log records each data product being stored and storage status of each storage operation. Storage failure results in a rejection message being sent to the requester. The rejection message is entered in the archive activity log and forwarded to the Data Archive Manager and the operator for corrective action. (The most likely reason for a storage failure is a physical hardware problem.) The Data Storage Request continues or is discarded in accordance with DAAC Policy. The checksum value, storage status, and other selected metadata is forwarded to the Science Data Server in a status message upon completion of the Data Storage Request.	Store data granules in the permanent archive.
3		The metadata is updated with a unique granule ID and names of the science data files archived.	Prepare the metadata for insertion in the database.
4		Science Data Server receives and logs the Data Storage Request status message from Storage Management. Validation failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message and the Data Storage Request Status Message are forwarded to QA for evaluation and correction. The Data Insert Request continues or is discarded in accordance with DAAC Policy. The Data Server produced metadata update file is loaded into the metadata database. The status of the metadata load is entered in the event log. Load failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message and the Data Storage Request Status Message are forwarded to QA for evaluation and correction. The ESDT Insert Request continues or is discarded in accordance with DAAC Policy.	Store metadata.
5		The Science Data Server logs completion of the ESDT Insert Request in the event log and reports completion of the ESDT Insert Request to the Data Archive Manager, the operator console and to the insert Requester (the Ingest Subsystem in this case). Each of the above entities would also be notified if the request failed and reason(s) for failure is/are identified.	Report ESDT Insert Request Status.
6		The Science Data Server will then examine the event list for all subscriptions for that event. Subscription notifications are sent to the appropriate entities as appropriate and distribution processing is initiated.	Process subscriptions based on newly inserted data.

Table 4.2.2.8-2. ESDT Delete Scenario

Purpose and Precondition:

The Data Server is required to indicate that the ESDT has been deleted. This requires an update to the metadata for the ESDT to indicate that the data is no longer available. Existing UR's and granule IDs will be able to access the data, however no new references to the data will be created past the time that the ESDT has been marked as deleted.

The metadata is updated to reflect that the ESDT has been deleted. The files associated with the granule ID are not actually deleted from the permanent archive (i.e., the archive is truely permanent).

Step	Operator/User	System	Purpose
1	Note: All of the messages listed in each Data Server Subsystem scenario under the "System" heading may be viewed by operations personnel by monitoring the display or by browsing the log files provided by MSS.	The Ingest Subsystem sends a ESDT Delete Request to the Science Data Server. Receipt of the Request is logged, and a request identifier is associated with the ESDT Delete Request. The content of the request is validated. Validation success results in the request being queued for later processing.	Initiate session between the Ingest Subsystem and a Data Server.
2		The metadata is updated with a delete indicator.	Prepare the metadata for insertion in the database.
3		The Data Server produced metadata update file is loaded into the metadata database. The status of the metadata load is entered in the event log. Load failure results in a rejection message being sent to the requester. The rejection message is entered in the event log and forwarded to the Data Archive Manager and the operator. The rejection message is forwarded to QA for evaluation and correction. The ESDT Delete Request continues or is discarded in accordance with DAAC Policy.	Store metadata.
4		The Science Data Server logs completion of the ESDT Delete Request in the event log and reports completion of the ESDT Delete Request to the Data Archive Manager, the operator console and to the deletion Requester (the Ingest Subsystem in this case). Each of the above entities would also be notified if the request failed and reason(s) for failure is/are identified.	Report ESDT Delete Request Status.